The NIBM Student Management System

The NIBM Student Management System is an advanced database designed to optimize campus administration. Its primary purpose is to streamline critical tasks and enhance data management for the entire institution.

Key Objectives

1 Centralized Data Storage

Consolidating all student information in a single, secure repository for easy access and management.

2 Enrollment Tracking

Efficiently managing student enrollment, tracking course selections, and monitoring class capacity.

3 Grade Recording and Reporting

Precisely storing and reporting student grades, facilitating academic progress tracking and reporting.

4 Attendance Monitoring

Streamlining attendance tracking for students and faculty, improving accountability and transparency.

Database Design Approach

1

Requirements Gathering

Identifying the specific data needs of the institution, including student details, course information, and administrative requirements.

2

Data Modeling

Creating a logical representation of the data relationships, ensuring data integrity and consistency.

3

Schema Development

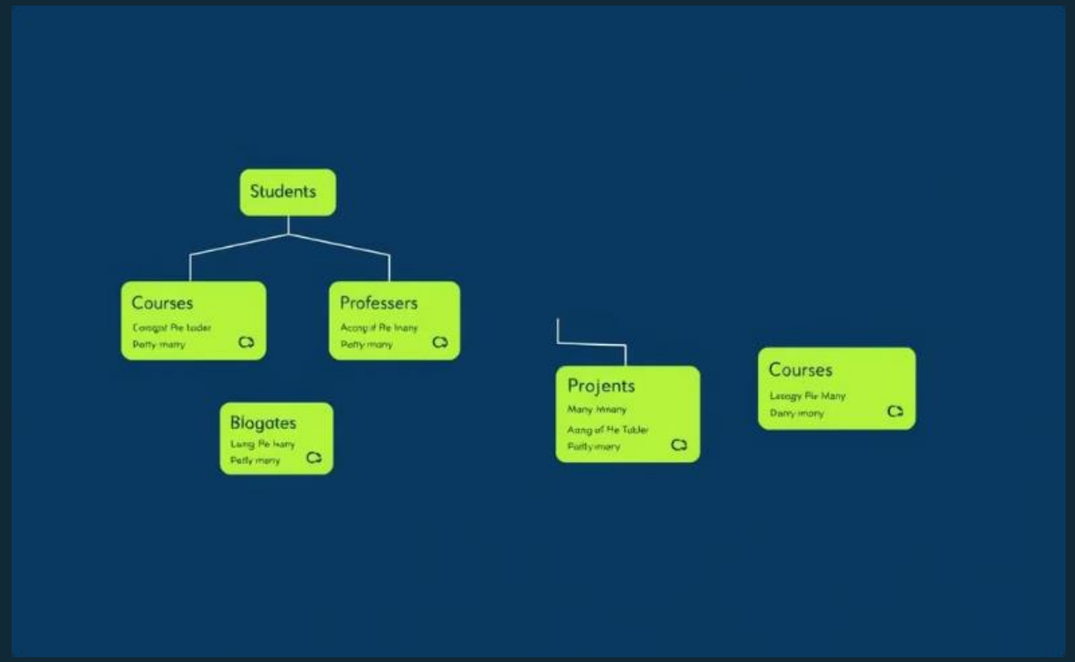
Translating the data model into a relational schema, defining tables, columns, and relationships.

4

Implementation

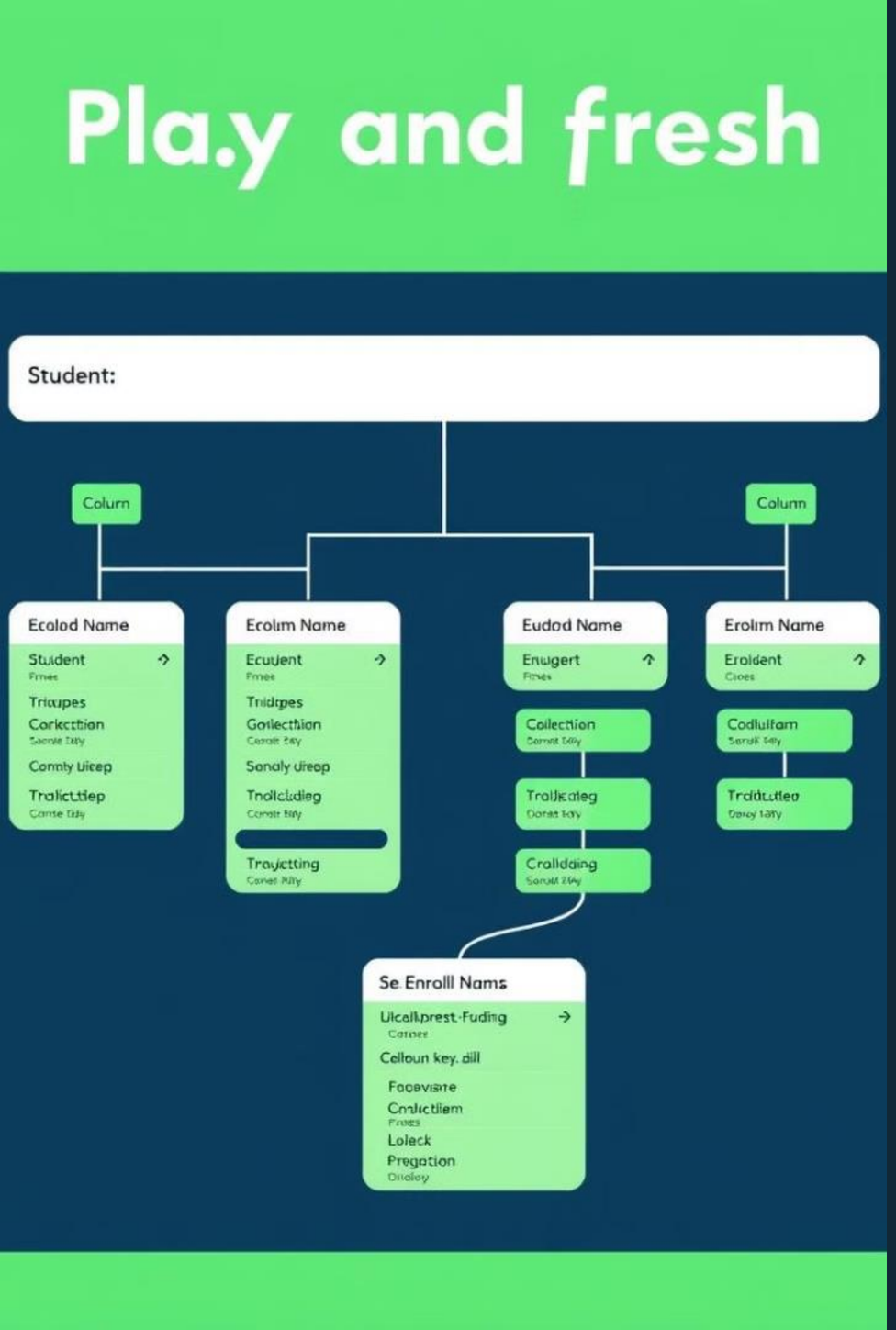
Creating the actual database structure and tables within the chosen database management system (MS SQL Server).

Conceptual Design: ER Diagram



Entity Relationship Diagram (ERD)

This visual representation outlines the entities within the system (e.g., Students, Courses) and their relationships (e.g., Enrollment).



Logical Design: Relational Schema

Table Name	Columns	Data Types
Student	StudentID, FirstName, LastName, Email, Phone, Address	INT, VARCHAR, VARCHAR, VARCHAR, VARCHAR, VARCHAR
Course	CourseID, CourseName, Department, InstructorID, Credits	INT, VARCHAR, VARCHAR, INT, INT
Enrollment	EnrollmentID, StudentID, CourseID, Grade	INT, INT, INT, VARCHAR

Physical Design: Database Creation

Database Creation

The physical database is created using SQL Server Management Studio, specifying the database name and other properties.

Table Creation

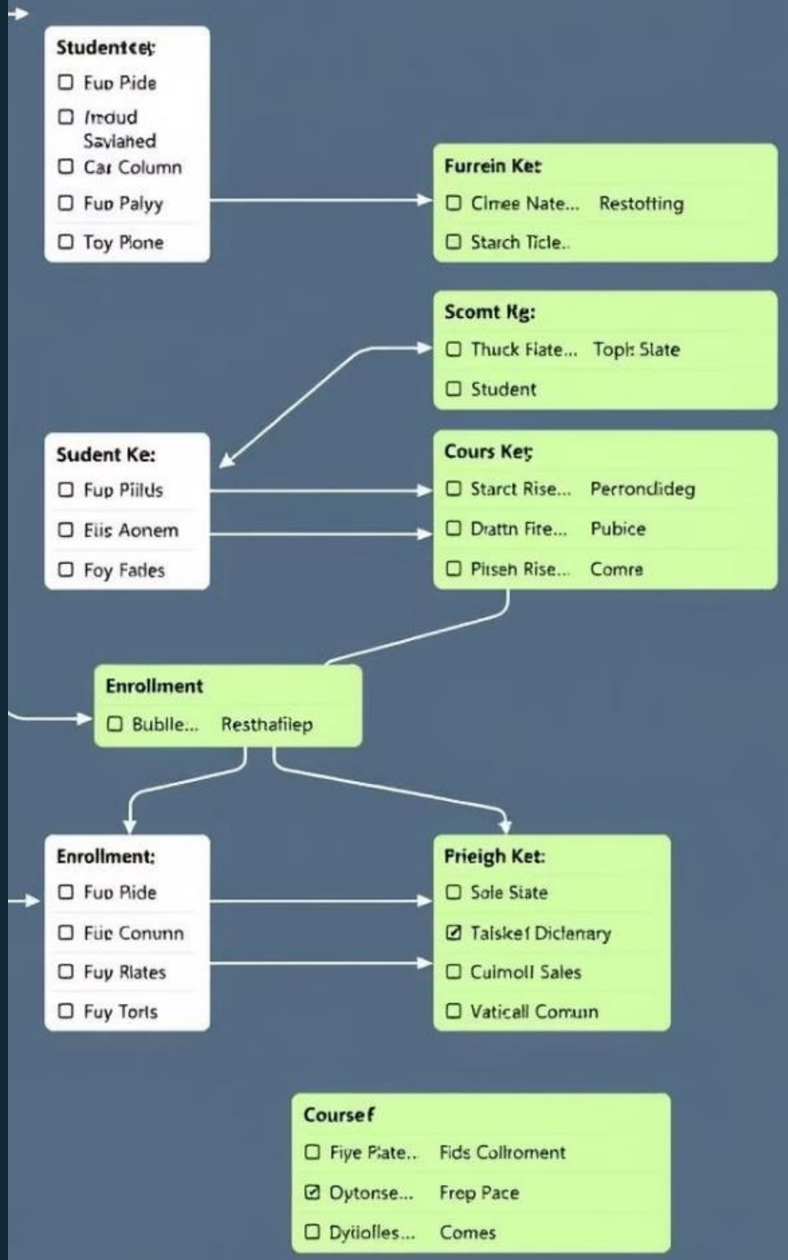
SQL statements (CREATE TABLE) are used to define the structure of each table, including column names, data types, and constraints.

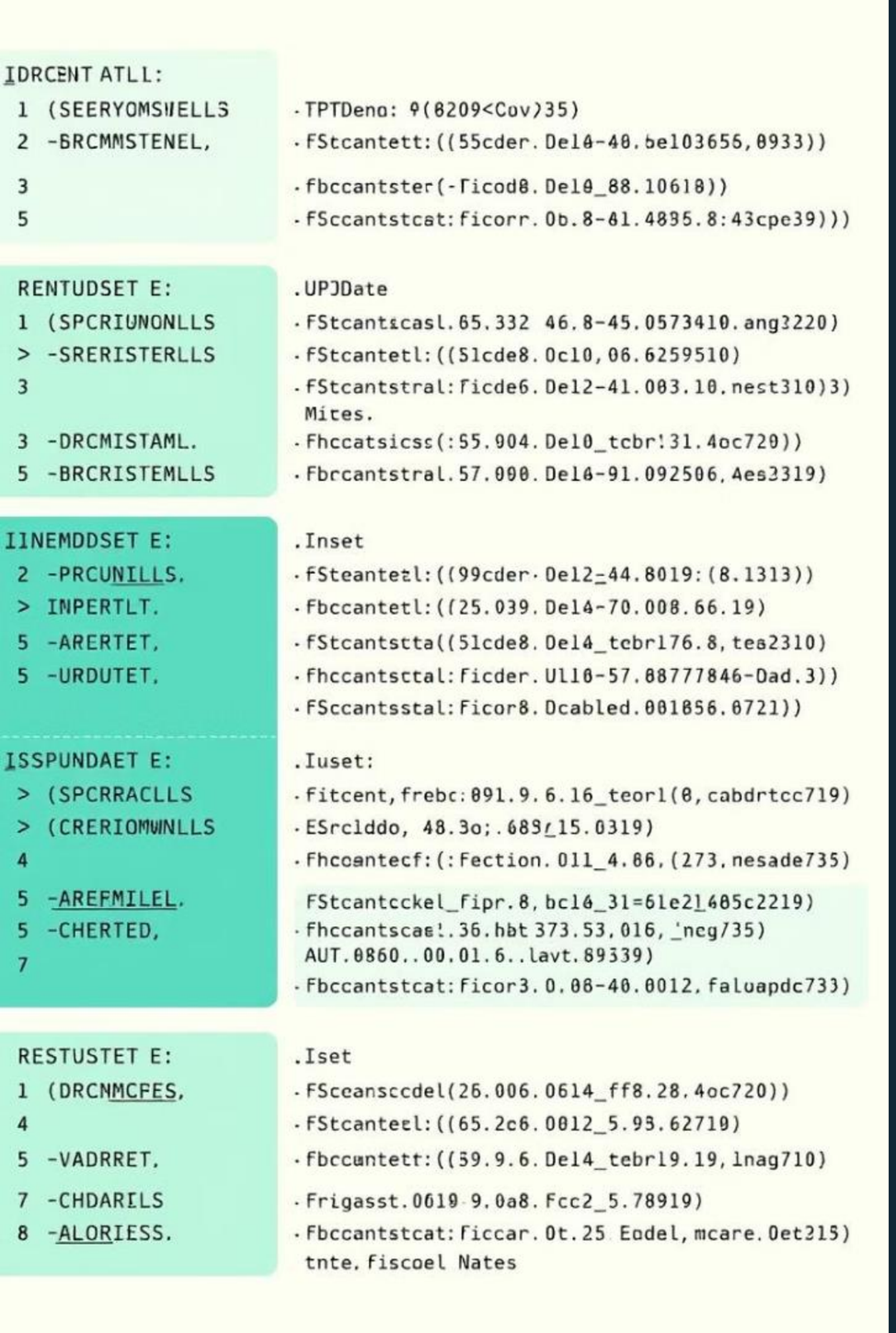
Relationship Establishment

Foreign key constraints are defined to link tables based on relationships specified in the ERD and relational schema.

Table Creation and Constraints

```
CREATE TABLE Student (
    StudentID INT PRIMARY KEY,
    FirstName VARCHAR(255),
    LastName VARCHAR(255),
    Email VARCHAR(255),
    Phone VARCHAR(20),
    Address VARCHAR(255)
);
CREATE TABLE Course (
    CourseID INT PRIMARY KEY,
    CourseName VARCHAR(255),
    Department VARCHAR(255),
    InstructorID INT,
    Credits INT
);
CREATE TABLE Enrollment (
    EnrollmentID INT PRIMARY KEY,
    StudentID INT,
    CourseID INT,
    Grade VARCHAR(2),
    FOREIGN KEY (StudentID) REFERENCES Student(StudentID),
    FOREIGN KEY (CourseID) REFERENCES Course(CourseID)
);
```





Data Manipulation (DML Statements)

INSERT

Used to add new records to tables. For instance, adding a new student or enrolling a student in a course.

UPDATE

Used to modify existing records. For example, updating a student's address or changing the grade for a course.

DELETE

Used to remove records from tables. This could involve removing a student from the system or deleting a course.

Data Control (DCL Statements)

1

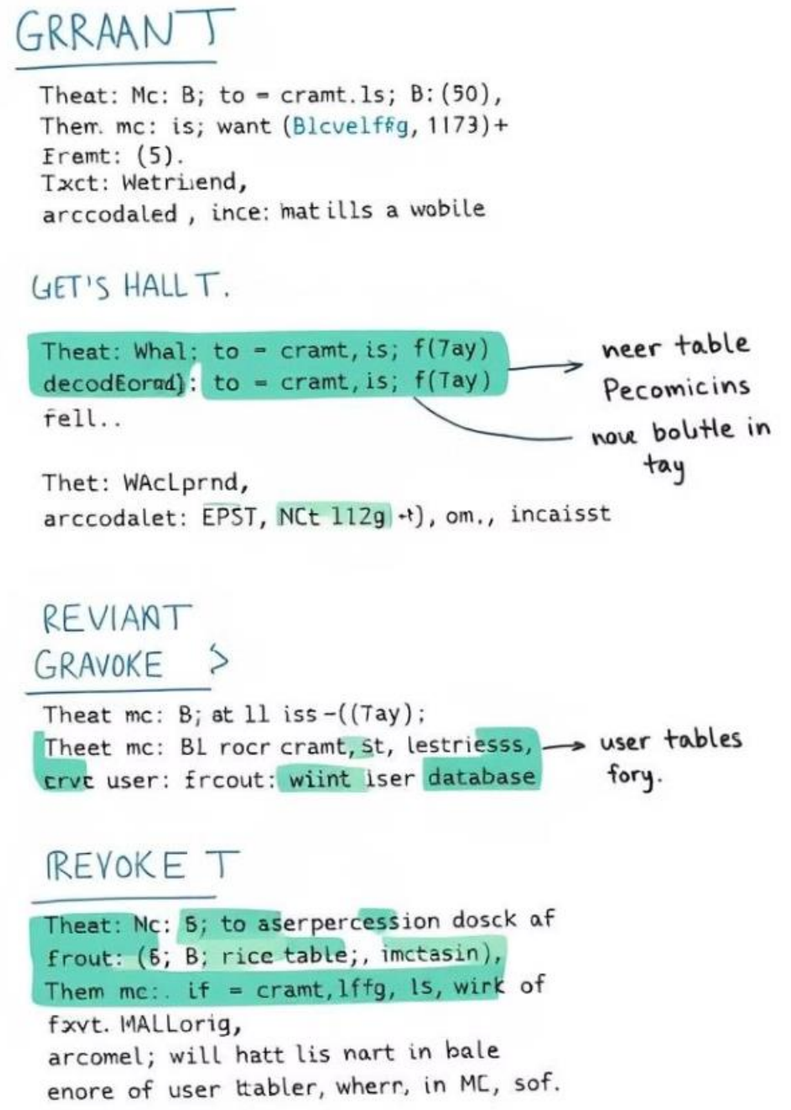
GRANT

Used to assign specific permissions to users or roles. This enables controlled access to data based on user roles and responsibilities.

2

REVOKE

Used to remove permissions previously granted to users or roles, ensuring data security and integrity.



Advantages and Conclusion

Data Integrity and Consistency

Relational database design ensures data integrity through constraints, preventing data redundancy and inconsistencies.

Reporting and Decision-Making

The database facilitates data analysis and reporting, enabling administrators to make informed decisions based on accurate data.

Scalability and Adaptability

The system can be easily scaled to accommodate growth in student enrollment and course offerings, ensuring long-term sustainability.

Security and Access Control

MS SQL Server offers robust security features, allowing granular control over user permissions and data access.